Skin-Color based Particle Filtering for Human Face Tracking

Tianrui WU, Yuexian ZOU and Wei WANG
Key Laboratory of Integrated Microsystems
Shenzhen Graduate School of Peking University
Shenzhen, 518055, China
wutianrui@pku.edu.cn, zouyx@szpku.edu.cn

***Abstract*—Skin color is a very important feature for the real time face tracking. By analyzing the skin color distributions of different people, we propose a novel face tracking algorithm which integrates the chromatic color information of the face skin region into the particle filtering (PF) framework. With the assumption of the Gaussian distribution of the face chromatic color, a Gaussian model is used to project the chromatic information of the YC_bC_r face image into a chromatic probability gray image. The histogram of the chromatic probability gray image is considered as the observation model for the PF. The update of the weight vector of the PF is determined by the Bhattacharyya distance between the reference model and the measured observed model. Extensive experiments have showed that our proposed algorithm performs quite well under the varying illumination, the full occlusion with the complex video background in terms of the tracking ability.**

***Keywords*—particle filtering, face tracking, skin-color Gaussian model**

I. INTRODUCTION

As an active research topic in computer vision, automatic human face tracking has attracted intensive research and been rapidly developed with the speedy growing of computer technology and the high demand of the commercial applications. This technology has a wide application in intelligent security control and perceptual user interfaces, etc [1]. Although research results showed that the significant achievements have been achieved in this area, there are still some technique difficulties remained unsolved. The face tracking under complex environment, varying illumination and full occlusion is one of them [2].

Skin color has been proven to be a very effective feature in the video-based face tracking technique. Without doubting different people have different skin colors, some research studies revealed that the major difference mainly lies in the intensity rather than the chrominance of the face images [3]. Research results also showed that face trackers using the skin color are robust to partial occlusion and changing scale [4]. In 1998, Bradski proposed a face tracker based on the mean shift algorithm as CAMSHIFT [5], which used the color histogram to model the object's color. CAMSHIFT is able to track the face objects in real time manner; however, it can not work well for the clutter or full occlusion situation.

In the field of video-based tracking, particle filters (PF) provide a practical way for solving non-Gaussian, nonlinear tracking tasks. Recently Nummiaro[6] and Pérez[7] developed an object tracker by integrating the histogram based color probability model into the particle filtering framework respectively. Research results showed that this approach provides a new method to solve the face tracking problem under the full occlusion and complex environment.

In this paper, we carefully studied the distribution of the chromatic information of the face images of different people, the chromatic information extraction method, and the modeling of chromatic information distribution. At the end, we proposed a new approach to integrate the chromatic information of the face image into the particle filtering framework. In order to make the comparison study, the conventional face tracking algorithm by using skin-color based particle filtering is also implemented. With fully making use of the robustness and the insensitiveness of the face chromatic information under varying illumination, full occlusion condition as well as the complex background, the proposed algorithm is expected to perform more robust and efficient than the conventional skin-color based particle filtering algorithm.

In the remainder of this paper, we introduce the analysis of the face images and skin-color extraction approach in section 2. In section 3, skin color based particle filter tracking algorithm will be described in details. Section 4 will present some experimental results and finally the conclusion is given in section 5.

II. SKIN-COLOR EXTRACTION

To segment the face region from the video image is an important task for the face tracking. Since the skin color is a unique feature of the face and using the skin color information to extract the face region from the background is a commonly used approach. In order to achieve this, a good skin color model that is suitable for different skin colors under different lighting conditions is desired [3]. In the common RGB space, the triple component (R, G, B) represents not only chromatic component but also luminance component, which may vary

978-1-4244-1674-5/08 /$25.00 ©2008 IEEE

across a person's face due to the ambient lighting, and it is not a reliable measure in separating skin from non-skin region.

It is noted that, in the YC_bC_r space, luminance can be fully separated from the chromatic components. The chromatic colors, also known as "pure" colors in the absence of luminance, are defined by C_b and C_r.

In order to evaluate the distribution of the chromatic components in YC_bC_r space, a total of 200 different face images (100×100 pixels each sample, refer to Fig.1) are selected. Without loss of the generality, we chose the face samples taken from persons of different ethnicities: Asian, Latino, Caucasian and African. For reducing the affects of the noises on the chromatic information distribution, a low-pass filter is applied on each sample face, where the impulse response of the low-pass filter is given by [9]

$$\frac{1}{9}\begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix} \tag{1}$$

A 2D histogram of all the samples is calculated and shown in Fig.2. It is clear to see that the chromatic color distribution of face images of different people is clustered in a small area and nearly unimodal-like. In other word, although the skin colors of different people appear to vary over a wide range, but the distribution of the chromatic components only varies in a limited range as shown in Fig.2. With this finding, it seems reasonable to approximate the skin color distribution by a Gaussian distribution $\mathbb{N}(\mu, \Sigma^2)$ [10], where the mean

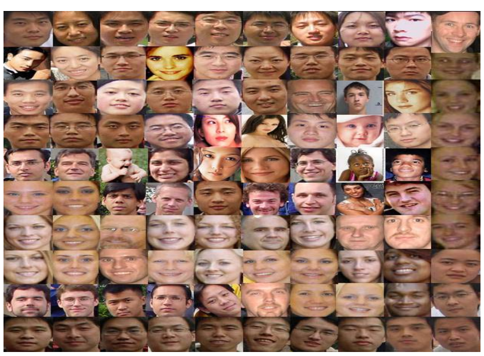


Figure 1. Part of the human face samples

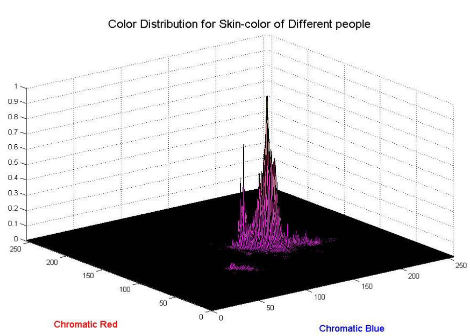


Figure 2. Skin-color distribution of different people

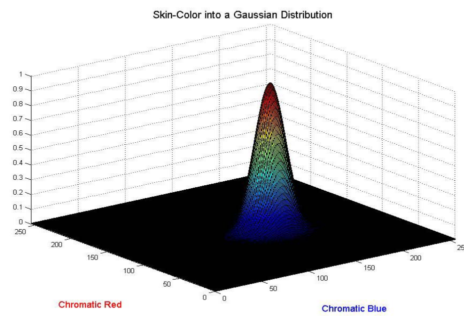


Figure 3. Skin-color Gaussian model $\mathbb{N}(\mu, \Sigma^2)$

$$\mu = (\overline{b}, \overline{r})^{\mathrm{T}} \tag{2}$$

and covariance

$$\Sigma^2 = \begin{bmatrix} \sigma_{bb} & \sigma_{br} \\ \sigma_{rb} & \sigma_{rr} \end{bmatrix} \tag{3}$$

with $\overline{b} = \frac{1}{N}\sum_{i=1}^{N} b_i$, $\overline{r} = \frac{1}{N}\sum_{i=1}^{N} r_i$, N is the number of the pixels of all samples ($=200 \times 100 \times 100$). And in this paper, we get the values

$$\mu = (115.08, 146.04)^{\mathrm{T}}$$

and

$$\Sigma^2 = \begin{bmatrix} 101.51 & -14.93 \\ -14.93 & 204.89 \end{bmatrix}$$

We can now obtain the likelihood of skin for any pixel of an image (in Fig.3). Therefore, if a pixel x_m, having converted from RGB color space into chromatic color space, has a chromatic pair value of $m = (b_m, r_m)^T$, the likelihood of skin for this pixel can then be computed as follows

$$P(m \mid skin) = P(b_m, r_m) = \exp[-0.5(m-\mu)^{\mathrm{T}}(\Sigma^2)^{-1}(m-\mu)] \tag{4}$$

Hence, this model can convert a color image into a gray scale image, called chromatic probability gray image, such that the gray value at each pixel shows the likelihood of the pixel belonging to the skin. Fig.4 shows the process of this conversion, and it's obviously that the skin regions are brighter than the other parts of the images.

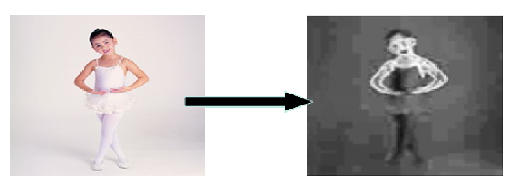


Figure 4. Color image and corresponding chromatic probability gray image

III. SKIN COLOR BASED PARTICLE FILTER TRACKING

In this section we'll discuss the dynamic model and the observation model used in the face tracking application, and how to incorporate the chromatic color of one image into the particle filter framework.

For the sake of completeness and to facilitate our derivation, we briefly review the setup of nonlinear filtering and particle filters in this section. Considering a discrete dynamics system, $\{\boldsymbol{x}_k, k\in\mathbb{N}\}$ and $\{\boldsymbol{y}_k, k\in\mathbb{N}\}$ denote the hidden states and the observations of the system at time k respectively. Because the posterior probability density function $p(\boldsymbol{x_k} \mid \boldsymbol{y_{1:k}})$ embodies all available statistical information, estimating $p(\boldsymbol{x_k} \mid \boldsymbol{y_{1:k}})$ is the objective of the tracking problem. Via the Chapman- Kolmogorov equation and Bayes' rule , we can get

$$p(\boldsymbol{x_k} \mid \boldsymbol{y_{1:k-1}}) = \int_{\boldsymbol{x}_{k-1}} p(\boldsymbol{x_k} \mid \boldsymbol{x_{k-1}}) p(\boldsymbol{x_{k-1}} \mid \boldsymbol{y_{1:k-1}}) d\boldsymbol{x_{k-1}} \tag{5}$$

$$p(\boldsymbol{x_k} \mid \boldsymbol{y_{1:k}}) = \frac{p(\boldsymbol{y_k} \mid \boldsymbol{x_k}) p(\boldsymbol{x_k} \mid \boldsymbol{y_{1:k-1}})}{p(\boldsymbol{y_k} \mid \boldsymbol{y_{1:k-1}})} \tag{6}$$

where $\boldsymbol{y_{1:k}} = \{\boldsymbol{y_k}, i=1,\dots,k\}$, $p(\boldsymbol{x_k} \mid \boldsymbol{x_{k-1}})$ and $p(\boldsymbol{y_k} \mid \boldsymbol{x_k})$ represent the system's dynamic model and likelihood which are often non-Gaussian. So in a PF frame work, the posterior distribution can be approximated by a set of weighted points $\{x_k^m, m=1,\dots,M\}$

$$p(\boldsymbol{x_k} \mid \boldsymbol{y_{1:k}}) \approx \sum_{m=1}^{M} w_k^m \delta(\boldsymbol{x_k} - \boldsymbol{x}_k^{\boldsymbol{m}}) \tag{7}$$

ensures that $\sum_{m=1}^{M} w_k^m = 1$, and the importance weight w_k^m is given as

$$w_k^m = w_{k-1}^m \frac{p(\boldsymbol{y_k} \mid \boldsymbol{x_k^m}) p(\boldsymbol{x_k^m} \mid \boldsymbol{x_{k-1}^m})}{q(\boldsymbol{x_k^m} \mid \boldsymbol{x_{k-1}^m}, \boldsymbol{y_{1:k}})} \tag{8}$$

Equation (8) is the particle weight updating equation. The particles are drawn from an importance density as $\boldsymbol{x_k^m} \sim q(\boldsymbol{x_k^m} \mid \boldsymbol{x_{0:k-1}^m}, \boldsymbol{y_{1:k}})$. In some PF algorithms, i.e. the bootstrap algorithm [8], the dynamic model $p(\boldsymbol{x_k^m} \mid \boldsymbol{x_{k-1}^m})$ is used as the importance density since it is independent of the measurement information, and simple to implement as well. In that case, (8) comes to

$$\tilde{w}_k \approx \tilde{w}_{k-1} p(\boldsymbol{y_k} \mid \boldsymbol{x_k}) \tag{9}$$

From (9) we can find that designing the likelihood $p(\boldsymbol{y_k} \mid \boldsymbol{x_k})$ is the key of the weight updating and the topic of this paper. In the following subsection, we'll show how the chromatic color can be used to generate the better likelihood and the procedure of the particles' updating.

A. *State space and Dynamic model*

As for video-based face tracking, the objective is marking the human face in each video frame quickly and precisely, and recording the track of the moving face. We represent the object by a rectangle window, whose location at time k is specified by the $d_k = (P_{X,k}, P_{Y,k})$ in the image coordinate system and scales are $(H_{x,k}, H_{y,k})$. So a hypothesized state at time k is given as

$$\boldsymbol{x_k} = [P_{x;k}, P_{y;k}, H_{x;k}, H_{y;k}] \tag{10}$$

For the randomicity of face movement, as described in [2][6][7], the dynamic model could be learned from a set of pre-labeled training sequences, and in our application we just model the dynamics as a simple second-order AR process

$$\boldsymbol{x_{k+1}} - \boldsymbol{x_k} = \boldsymbol{x_k} - \boldsymbol{x_{k-1}} + v_k \tag{11}$$

where $v_k \sim N(0,\Sigma)$ is a zero-mean Gaussian stochastic component.

B. *Observation model*

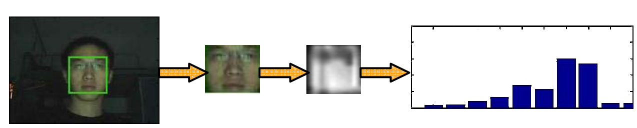
Instead of modeling target by color information without separating the luminance and considering the features of facial skin color [6][7], we firstly convert the raw image of each frame to a chromatic probability distribution image via the skin-color Gaussian model, then adopt the histogram of the chromatic probability gray image as our observation model.

For each sample, the region of the rectangle bounding the face at time $k, R(\boldsymbol{x_k})$, is converted into YC_bC_r space and further the chromatic probability gray image $R_g(\boldsymbol{x_k})$ is obtained. We calculate the histogram $\{h_k(n;\boldsymbol{x_k})\}_{n=1,2,\dots N}$ of the image $R_g(\boldsymbol{x_k})$ and acquire

$$h_k(n;x_k) = K \sum_{u\in R_g(x)} \delta[n_k(u) - n] \tag{12}$$

where δ is the Kronecker delta function, K is a normalization constant and $n_k(u) \in \{1,\dots,N\}$ assigns one of the N-bins of the histogram to a given gray value at pixel location u in frame k. Fig.5 shows the process of computing the histogram.

At time k, the observation model $h_k(\boldsymbol{x_k})$ associated with a hypothesized state $\boldsymbol{x_k}$ will be compared to the reference



(a) (b) (c) (d)

Figure 5. (a) Face object with rectangle window in frame k. (b) The face region. (c) Corresponding chromatic probability gray image. (d) The histogram of the gray image(c).

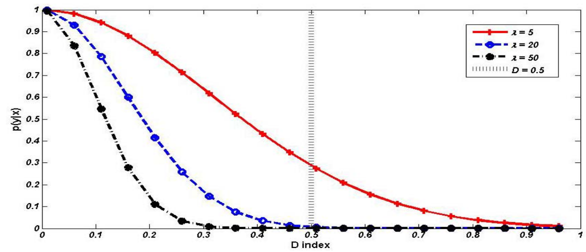


Figure 6. The influence of the choice of λ on distribution $p(\boldsymbol{y}_k \mid \boldsymbol{x}_k)$

model $h^*(\boldsymbol{x}_{k_0})$ which we gather at time k_0 and a similarity distance is used to measure them.

A popular measure between two distributions is the Bhattacharyya coefficient [13]. Considering discrete densities such as two gray histograms $h_k(\boldsymbol{x}_k)$ and $h^*(\boldsymbol{x}_{k_0})$, the coefficient is defined as

$$D[h^*(n;\boldsymbol{x}_{k_0}),h_k(n;\boldsymbol{x}_k)]=\left[1-\sum_{n=1}^{N}\sqrt{h^*(n;\boldsymbol{x}_{k_0})h_k(n;\boldsymbol{x}_k)}\right]^{\frac{1}{2}} \quad (13)$$

The smaller D is, the more similar two distributions are.

For two identical histograms, we obtain D=0 indicating a perfect match. And the data likelihood must favor candidate histograms close to the reference histogram, so referred to [7], we count the likelihood of observation model as

$$p(\boldsymbol{y}_k \mid \boldsymbol{x}_k)=\exp\{-\lambda\cdot D^2[h^*(n;\boldsymbol{x}_{k_0}),h_k(n;\boldsymbol{x}_k)]\} \quad (14)$$

where λ is a constant. In order to confirm its value, we suppose values {D} as a sequence of length 20 in the range [0,1], shown in Fig.6, and draw three curves indicating the likelihood distribution $p(\boldsymbol{y}_k \mid \boldsymbol{x}_k)$ when λ equates 5, 20 and 50(red, blue and black respectively). If we set D=0.5 as the threshold (grey dashed), which means the defined function (14) just make effect in the condition D<0.5, and when $D \geq 0.5$ the likelihood is zero, from Fig.6, we find the blue curve indicating λ=20 satisfies our need well. Therefore, we confirm the value of λ equate 20.

Thus by substituting (14) into (9), the weight updating formula is

$$w_k \approx w_{k-1}\exp\{-\lambda\cdot D^2[h^*,h_k]\} \quad (15)$$

C. The PF tracking using skin color

Given the rectangle window bounding face, the sample set $\{\boldsymbol{x}^m_{k+1}, m=1,\ldots,M\}$ at time k+1 can be propagated using (11) and the synoptic algorithm describes as follows.

1) Initial state: $\boldsymbol{x}_{k_0}=[P_{X,k_0},P_{Y,k_0},H_{x,k_0},H_{y,k_0}]$, with weights $w^m_0 = 1/M$, set k=0.

2) At time k, draw $\{\hat{\boldsymbol{x}}^m_{k+1}, m=1,\ldots,M\}$ from second order AR dynamics using (11).

3) Extract the sample's candidate face region, and convert it into a chromatic probability gray image using (4).

4) Computation of candidate histogram: for m=1...M, compute $q_k(n;\boldsymbol{x}_k)$ according to (12).

5) Compute the Bhattacharyya distance $D[q^*,q_k]$ according to (13).

6) Obtain the sample's importance weight at time k+1 $\tilde{w}^m_{k+1}=\exp\{-\lambda\bullet D^2[h^*,h^m_{k+1}]\}$ from (15).

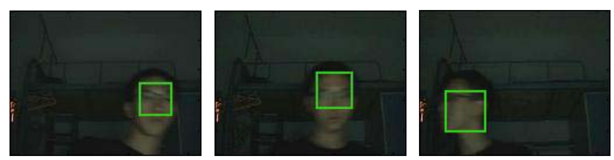
7) Calculate the mean state of the set $\{\hat{\boldsymbol{x}}^m_{k+1}, m=1,\ldots,M\}$: $\mathrm{E}[\{\hat{\boldsymbol{x}}^m_{k+1}\}]=\sum_{m=1}^{M}\tilde{w}^m_{k+1}\hat{\boldsymbol{x}}^m_{k+1}$,which describes the center location and scale of the rectangle.

8) Resample the set $\{\hat{\boldsymbol{x}}^m_{k+1}, m=1,\ldots,M\}$ to get the updated set $\{\boldsymbol{x}^m_{k+1}, m=1,\ldots,M\}$ in terms of the importance weights $\{\tilde{w}^m_{k+1}, m=1,\ldots,M\}$, referred to , and then assign the new weights: $w^m_{k+1} = 1/M$.

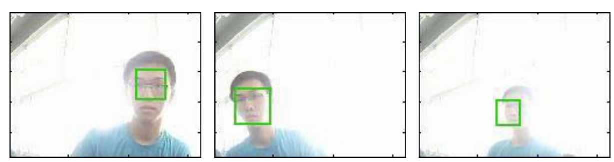
9) Set k=k+1, and go back to 2).

IV. EXPERIMENT RESULTS

To evaluate the effectiveness of our algorithm, a series of experiments are designed. Test video sequences S1, S2, S3 and S4 are 320×240 in resolution and captured at 20 frames per



(a)Weak lighting environment (video sequences S1)



(b)Strong lighting environment (video sequences S2)

Figure 7. Illustration of our proposed algorithm in the dark and bright environment respectively

#49 #50 #51 #52

#53 #54 #55 #56

Figure 8. Illustration of face tracking in full occlusion and complex environment (video sequence S3)

second in the complex lab environment. The initial state of human face is preset. Experiments are carried out on a PC with a 2.8GHz Celeron(R) CPU and 1G memory, using Matlab 7.0 and VC++ 6.0.

In some experiments, we'll illustrate the differences between the proposed skin color based PF tracker and the classical color-based particle filter tracker in [7]. As for the bin number of histogram, we used N=100 in the reference method, and N=10 in our method. The sample numbers are both 50 in two methods.

A. Varying illumination

We captured the video sequence S1 and S2 in the dark and strong ambient lighting respectively. Fig.7(a) and Fig.7(b) illustrate three frames with moving face, together with the face tracking results using our proposed algorithm, respectively. It can be seen from Fig.7(a) that our proposed tracker can keep marking the moving face quite well when the face ambient is dark. Moreover, we can conclude that the proposed tracker can track the face in the high luminance robustly, even the vague face in the third figure of Fig.7(b).

B. Occlusion

In this experiment, the proposed method is applied for the face tracking in full occlusion and complex environment. And test sequence S3 was captured in the complex background with skin-colored, multi-lighting and cluttered texture. The test results are displayed in Fig.8. As shown in frame 51~53, when the face is completely occluded by the hand (also is skin-colored), our proposed tracker loses the object temporarily. When the hand moves away, our tracker can find the face quickly (one frame later in #55). Hence, the results demonstrate that our method is robust in full occlusion and complex environment.

C. Face rotation

In this section, a tracking performance comparison between our proposed method and the color-based particle filter tracker [7] will be carried out using video sequences S4. S4 video was captured in a laboratory with a complex background, in addition, in the video a human face rotated from the left to the right with a slight scale change. In this experiment, 60 frames (#170~#230) are used.

The 6 frames in the top and the bottom rows of Fig.9 illustrate the face tracking results of our proposed algorithm and the one proposed in [7], respectively. From the top row, the result shows that our proposed tracker can explicitly rectangle the target face object and scale corresponding to the change of the target face object. However, from the bottom row, it is clear to see that face tracker [7] cannot rectangle the target face well. It behaves certain bias in terms of tracking and the scaling performance.

In order to illustrate the tracking trajectory of the two algorithms, the center positions of the tracking box are shown in the middle of Fig.1. The blue and red trajectories are the center positions of tracking box using the proposed algorithm and the algorithm in [7], respectively. It is clear to see that the dynamic range of the blue dots is much bigger than the red dots. This result tells us that the proposed algorithm is able to track face moving in a large range compared with the algorithm proposed in [7].

V. CONCLUSION

A novel method for tracking human face using chromatic color of skin region based particle filtering framework has been proposed. In our algorithm, each sample's candidate tracking region is converted into a chromatic probability gray image, and the histogram of this image is considered as the

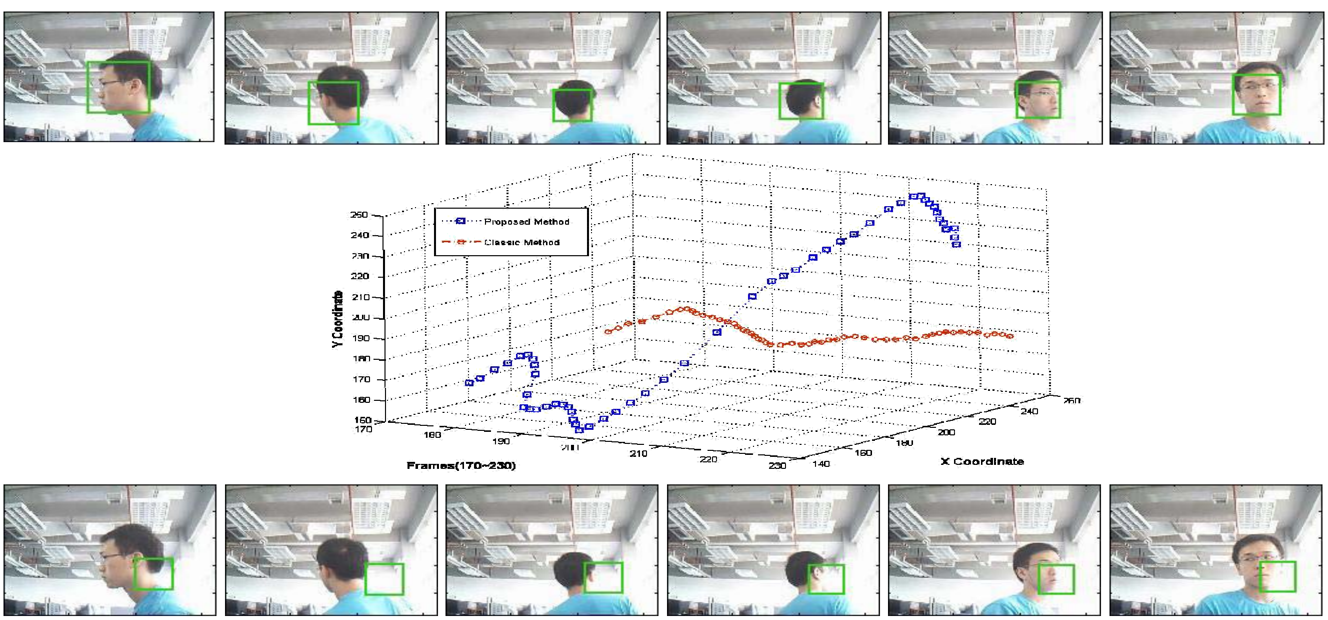


Figure 1 the top row are 6 frames result of our method used in video sequences S4 (face rotation and complex background) and the lower sequences are the ones with classic tracker[7]. The figure in the middle demonstrate those tracker's trajectories of the box in the frame 170~230 respectively. (the red one comes from the reference algorithm [7], and the blue one is by our algorithm) target model.Then by comparing it with a specific model using Bhattacharyya distance, we can make all the samples' weight updating and implement the predicting of new state. Extensive experiments are carried out in the real lab environment to evaluate the face tracking performance. It has shown that our proposed algorithm is able to track the face object in cluttered texture, multi-lighting, varying illumination, face rotation and full occlusion condition. This technique may provide an optional solution for face tracking applications in real-time and complicated environment, such as the surveillance system, etc.

REFERENCES

[1] Hu Weiming, Tan Tieniu, Wang Liang, Maybank S. "Survey on Visual Surveillance of Object Motion and Behaviors," IEEE Transactions on Systems, Man and Cybernetics, 34(3): pp.334–352, August, 2004.

[2] Qicong Wang and Jilin Liu. "The Improved Particle Filter for Object Tracking," Proceedings of the 6th World Congress on Intelligent Control and Automation, pp.10275–10279, vol.2, 2006.

[3] Yang M.H., Kriegman D.J., Ahuja N., "Detecting Faces in Images: A Survey [J]," IEEE Trans. on Pattern Analysis and Machine Intelligence, 2002, 24(1), pp. 34–58.

[4] D.Comaniciu, V.Ramesh, P.Meer, "Real-time tracking of non-rigid objects using mean shift," IEEE Conference on Computer Vision and Pattern Recognition, pp.142–149, vol.2, 2000.

[5] G.Bradski, "Computer vision face tracking for use in a perceptual user interface," Intel Technology Journal, vol 2nd Quarter, 1998.

[6] Nummiaro K, Koller-Meier E, Van Gool L. "Object tracking with an adaptive color-based particle filter."Symposium for Pattern Recognition of the DAGM, 2002. pp.353–360.

[7] Pérez P, Hue C, Vermaak J, Gangnet M. "Color-based probabilistic tracking." European Conference on Computer Vision, 2002, pp.661–675

[8] N.Gordon,D.Salmond,and A.F.M.Smith, "Novel approach to non-linear and non-Gaussian Bayesian state estimation," Proc. Inst. Elect. Eng., vol.140, pp.107–113, 1993.

[9] Rafael C. Gonzalez, Richard E. Woods. Digital Image Processing Second Edition. Prentice Hall, 2002

[10] Jie Yang and Alex Waibel. "A real-time face tracker."In Proceedings of WACV,pp.142–147, 1996.

[11] MS Arulampalam, S. Maskell, N. Gordon, et al. "A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking."IEEE Trans. Signal Processing, 2002, 50(2):,pp.174–188

[12] A. Doucet, N. de Freitas, and N. Gordon. An Introduction to Sequential Monte Carlo Methods, ser. Statistics for Engineering and Information Science. Springer, ch. 1, pp.3–14, 2001.

[13] Comaniciu D, Ramesh V, Meer P. "Kernel-based object tracking."IEEE Transactions on Pattern Analysis and Machine Intelligence, 2003, 25(5): 564–577.

This work is supported by National High Technology Research and Development Program of China (*863 Program, No.2006AA04Z2477 and 2007AA11Z224*)